Distribution-based objectives for Markov Decision Processes

S. Akshay, Blaise Genest, Nikhil Vyas

Distribution-based objectives for Markov Decision Processes

S. Akshay
Dept of CSE
Indian Institute of Technology Bombay
India
akshayss@cse.iitb.ac.in

Blaise Genest
Univ Rennes
CNRS, IRISA
France
blaise.genest@irisa.fr

Nikhil Vyas
EECS
MIT
USA
nikhilv@mit.edu

Abstract

We consider distribution-based objectives for Markov Decision Processes (MDP). This class of objectives gives rise to an interesting trade-off between full and partial information. As in full observation, the strategy in the MDP can depend on the state of the system, but similar to partial information, the strategy needs to account for all the states at the same time.

In this paper, we focus on two safety problems that arise naturally in this context, namely, existential and universal safety. Given an MDP $\mathcal{A}$ and a closed and convex polytope H of probability distributions over the states of $\mathcal{A}$, the existential safety problem asks whether there exists some distribution Δ in H and a strategy of $\mathcal{A}$, such that starting from Δ and repeatedly applying this strategy keeps the distribution forever in H. The universal safety problem asks whether for all distributions in H, there exists such a strategy of $\mathcal{A}$ which keeps the distribution forever in H. We prove that both problems are decidable, with tight complexity bounds: we show that existential safety is PTIME-complete, while universal safety is co-NP-complete.

Further, we compare these results with existential and universal safety problems for Rabin's probabilistic finite-state automata (PFA), the subclass of Partially Observable MDPs which have zero observation. Compared to MDPs, strategies of PFAs are not state-dependent. In sharp contrast to the PTIME result, we show that existential safety for PFAs is undecidable, with H having closed and open boundaries. On the other hand, it turns out that the universal safety for PFAs is decidable in EXPTIME, with a co-NP lower bound. Finally, we show that an alternate representation of the input polytope allows us to improve the complexity of universal safety for MDPs and PFAs.

1 Introduction

Markov decision processes (MDPs) are a basic model for stochastic dynamical systems combining probabilistic moves with non-deterministic choices. They find applications in various domains, such as control theory, AI, networks, verification, and so on. Theoretical study of MDPs has been focused on either qualitative (e.g. almost-sure properties) or quantitative questions on the behavior of the MDPs. A classical question is whether there exists a strategy to resolve the non-deterministic choices, under which the behavior of the stochastic system underlying the MDP satisfies or optimizes a given objective, often maximizing rewards or satisfying constraints. There are efficient algorithms in many of these cases and considerable work has gone into making them scale in practice.

On the other hand, in the presence of partial observation, i.e., when some of the states are indistinguishable, it is known that many of these results do not hold. Indeed, for partially-observable MDPs (POMDPs) and the so-called Rabin's probabilistic finite automata (PFAs), a zero-observation restriction where all states are indistinguishable, belief distributions (or belief states) need to be considered, at least indirectly. The belief distribution associates to each state the probability to be in that state according to the observations seen. Dealing quantitatively with the belief distribution is hard, and that is one of the reasons why many quantitative decision problems are undecidable for POMDPs and PFAs [16, 22].

In this paper, we take an alternate view of MDPs, which gives rise to an interesting trade-off between full observation and partial information. Using *distribution-based objectives*, we directly reason about the belief distribution. However, unlike partial information and as in fully observable systems, the strategy of the MDP can depend upon the state of the system. This view of MDPs has several related interpretations and applications, such as transformer of probability distributions [11]; and as described later below, in representing the evolution of a fluid population of agents.

Having fixed this view, we focus on (distribution-based) *safety* objectives. Our goal is to determine when we can control an MDP so that the belief distribution stays within a given safe convex region. More precisely, we consider the safe region to be given as a closed and convex polytope H over the set of distributions. We denote a strategy by σ, where at each time point i, $\sigma(i)$ chooses for each state a (distribution over) action(s). Once a strategy is fixed, the transformation between the belief distributions at time i and $i+1$ can be seen as a Markov chain $M_{\sigma(i)}$. We consider two questions in this setting: existential and universal safety. The question of *existential safety* asks whether there exists an initial distribution Δ in H and a strategy σ such that under σ, the belief distribution always remains in H, i.e., for all $n \in \mathbb{N}$, $\Delta \cdot M_{\sigma(1)} \cdots M_{\sigma(n)} \in H$. We also consider the dual question of *universal safety*, which asks if for all initial distributions in H, there is a strategy remaining in H.

Our main contributions, depicted in Table 1, are the following: we show that both the existential and universal safety problems are decidable for MDPs, and provide tight complexity bounds. First, we show that existential safety is PTIME-complete by showing that the safety problem over all time steps n can be reduced to the existence of a special distribution. For this, we use a strong fixed point theorem, namely the Kakutani fixed-point theorem. Hardness follows easily since the questions on convex polytopes capture

This work was partially supported by ANR projects STOCH-MC (ANR-13-BS02-0011-01), DST/CEFIPRA/Inria Associated team EQUAVE, DST/INSPIRE Faculty Award [IFA12-MA-17], Akamai Presidential Fellowship and NSF CAREER award CCF-1552651.

LICS '18, July 9–12, 2018, Oxford, United Kingdom

ACM ISBN 978-1-4503-5583-4/18/07…$15.00
https://doi.org/10.1145/3209108.3209185

Complexity of safety	MDPs	PFAs
Existential	PTIME-complete	Undecidable
Universal	co-NP-complete	EXPTIME and co-NP-hard

Table 1. A summary of the results in this paper (for polytopes under the H-representation)

linear programming. Next, we show that universal safety is co-NP-complete. Here the co-NP upper bound is obtained by using recent and state-of-the-art results from Quantified Linear Programming. However, hardness requires a complicated reduction.

In sharp contrast, we show that existential safety is undecidable for PFAs for H with closed and open boundaries, by a somewhat surprising reduction from the *universal* halting problem for 2-counter machines. On the other hand, it turns out that universal safety is still decidable for PFAs but with a complexity EXPTIME and is at least coNP-hard. These results hold when the polytope is given using equations, called H-representation. When polytopes are instead given using corner points, called V-representation, we can improve the complexity of universal safety to PTIME for MDPs and PSPACE for PFAs. This representation does not improve the complexity results for existential safety.

Before going to an example, we argue that these problems can be highly non-trivial. Let us consider the related problem of *initialized safety*, which asks whether there exists a strategy σ in the MDP such that from a given initial distribution $\Delta \in H$, the belief distribution produced by the strategy always remains in H, i.e., for all $n \in \mathbb{N}$, $\Delta \cdot M_{\sigma(1)} \cdots M_{\sigma(n)} \in H$. This initialized safety problem for MDPs trivially subsumes the initialized safety problem for Markov chains (by taking the size of the alphabet to be 1). Surprisingly, it turns out that this problem is already as hard as the Skolem problem [3], whose decidability is a long-standing open problem [29]. Only some subclasses are known decidable for arbitrary dimensions, such as ultimate-positivity (equivalent to an eventual safety condition) for restricted matrices where eigenvalues have multiplicity 1 [25]. The existential and universal safety problems can, respectively, be seen as under and over-approximations of the initialized safety problem. That is, if the existential safety problem has a negative answer, then so does the initialized safety problem, and the universal safety problem has a positive answer, then so does the initialized safety problem.

Motivating example As motivation, consider a population of yeasts under osmotic stress [23]. The stress level of the population can be studied through a protein which can be marked (by a chemical reagent). For the sake of illustration, consider the following simplistic model where a yeast can take 3 different discrete states, namely the concentration of the protein being high (state 1), medium (state 2) and low (state 3).

When a cell is on a saline substrate, it will evolve using one dynamics, described by the Markov chain M_{sa}, and when it is on a sorbitol substrate, it will evolve using another dynamics, described by the Markov chain M_{so}, given in Fig. 1. These two Markov chains give the proportion of the population of yeasts (considered as a fluid) moving from one protein concentration level to another, in one time step (say, 15 seconds) under this substrate. For instance, 20% of the yeasts with low protein concentration will have high protein concentration at the next time step under a saline substrate, which is represented by the value 0.2 in M_{sa}.

The difference between the MDP and the PFA model is that with the MDP model, the substrate may vary for each yeast, while for PFAs, there is a unique substrate for the whole population. We want to control this population of yeasts, to make it stay within some reasonable convex polytope H, e.g., the proportion of yeasts with high concentration of the protein (in state 1) stays inside the interval $[\frac{1}{4}, \frac{1}{2}]$. We can then ask two questions: whether for all initial configurations in H, there exists such a safe strategy, meaning that H is stable, and if not, whether there exists at least one initial configuration in H for which there is a strategy to stay inside H.

Related Work There has been considerable work concerning Markov Chains in the distribution-based context. As there is no choice of actions, this view coincides with unary PFAs. Further, the problem considered is to perform model-checking of distribution-based properties rather than strategy synthesis (there is no choice to resolve). In [6], it was shown that distribution-based properties cannot be expressed in the more classical probabilistic variant of the CTL* logic. In fact, these verification questions generalize the above mentioned initialized safety question and hence are also Skolem-hard for Markov chains [3]. However, one can find decidable subclasses as in [4], or approximate solutions for some distribution-based properties as in [1, 2] and also in [10], where the related isolation problem is tackled.

The existential safety problem has also been considered over *general real* matrices (rather than stochastic ones), in the special deterministic case (no control involved), where Tiwari [28] proved a PTIME algorithm for the case where the polytope is a half space using a fixed point approach similar to ours. However, that result uses the Brouwer's fixed point theorem, while ours needs the more powerful Kakutani's fixed point theorem as we have to deal with non-deterministic choices. More recently, a *continuous* version of existential safety has been proved decidable for another deterministic class (no control involved), namely Continuous Linear Dynamical Systems [24], using tools from Diophantine approximation.

Concerning non-deterministic systems (involving control) with distribution-based objectives, PFAs are a well-studied model. Quantitative questions are undecidable [7], as well as approximating quantitative questions [22]. Even some qualitative questions are undecidable, such as the value 1 problem [17], and only very restricted subclasses are known that ensure decidability of PFAs [11, 12, 16]. MDPs with the same semantics as we use have been compared with PFAs for the *qualitative* problem called almost-sure synchronization. This problem has been shown to be decidable in PSPACE for MDPs

$$\left(M_{sa} = \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.2 & 0.1 & 0.7 \end{pmatrix}, \quad M_{so} = \begin{pmatrix} 0.3 & 0.4 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \right)$$

Figure 1. Two actions sa, so and their Markov Chain effect

[14], while it is undecidable for PFAs [13], using a simple reduction to the undecidable reachability for PFAs. Recently, *qualitative* questions on PFAs presented as discrete (non-fluid) populations have been proved decidable, using results on parametric control [8]. Compared to these results, we show decidability of *quantitative* questions, namely existential and universal safety for MDPs.

Structure of the Paper In Section 2, we start by providing the definitions and notations for MDPs and PFAs. We also define the safety problems on convex polytopes and prove some preliminary results. In Section 3, we prove our first main result, namely PTIME-completeness of existential safety for MDPs. Section 4 is devoted to the undecidability of existential safety for PFAs. Sections 5 and 6 focus on decidability of universal safety for MDPs and PFAs respectively. Finally, in Section 7 we consider how the complexity is improved for polytopes given in the V-representation. Proofs omitted due to lack of space can be found in [5].

2 MDPs, PFAs and safety properties

In this section, we define Markov decision processes (MDPs) and probabilistic finite-state automata (PFAs) directly using a matrix notation. This corresponds to viewing MDPs and PFAs as transformers of probability distributions [11] rather than state transformers, and are equivalent to the common definition via transition systems.

Let $S = \{s_1, \ldots, s_n\}$ be a set of states, Σ a finite alphabet of actions. For all $1 \leq i \leq n$, we use $\vec{s_i}$ to denote the n-dimensional vector, which has 1 in position i and 0 elsewhere. We use Δ_1, Δ_2 etc. to denote arbitrary (probability) distributions over S, i.e., n-dimensional vectors $\Delta \in [0,1]^n$ such that $\sum_{i=1}^n \Delta(i) = \sum_{i=1}^n \vec{s_i} \cdot \Delta = 1$. We will sometimes use $|\cdot|_1$ to denote the ℓ_1-norm of a vector, i.e., sum of its entries. Thus for a distribution Δ, $|\Delta|_1 = 1$. Further, δ, δ' will denote sub-distributions over S, i.e., vectors from $[0,1]^n$, such that $|\delta|_1 \leq 1$. Similarly, we will use M, M' etc., to denote n-dimensional stochastic matrices (each row is a distribution). Any such matrix can be seen as defining the transition matrix of a Markov chain over the set of states S.

Definition 2.1. A *Markov decision process* or a *probabilistic finite-state automaton* is a tuple $\mathcal{A} = (S, \Sigma, (M_\alpha)_{\alpha \in \Sigma})$, where S is a set of states, Σ is the alphabet of actions, and $(M_\alpha)_{\alpha \in \Sigma}$ is a set of stochastic matrices, which will define how the probability mass in a state $s_i \in S$ is transformed playing any action $\alpha \in \Sigma$.

For instance, the motivating example is a PFA/MDP with $S = \{s_1, s_2, s_3\}$, $\Sigma = \{so, sa\}$, and M_{so}, M_{sa} as given in Fig. 1.

The difference between an MDP and a PFA is in the allowed one-step strategies (also called decision rules [26]). We start by defining one-step strategies of PFAs, which do not depend on the state:

Definition 2.2. A one-step strategy of a PFA $\mathcal{A} = (S, \Sigma, (M_\alpha)_{\alpha \in \Sigma})$ is a function $\tau : \Sigma \to [0,1]$ such that $\sum_{\alpha \in \Sigma} \tau(\alpha) = 1$. A one-step strategy τ is associated with the stochastic matrix:

$$M_\tau = \sum_{\alpha \in \Sigma} \tau(\alpha) M_\alpha$$

We now define the one-step strategies of an MDP, which may depend upon the state. For M_α a stochastic matrix, we denote by $M_{(\alpha,j)}$ the matrix obtained by taking M_α and setting all rows to be the 0-vector, except for the j-th row (associated with state s_j).

Definition 2.3. A one-step strategy of an MDP over S, Σ is a function $\tau : \Sigma \times S \to [0,1]$ such that for all $s \in S$, $\sum_{\alpha \in \Sigma} \tau(\alpha, s) = 1$. A one-step strategy τ is associated with the stochastic matrix:

$$M_\tau = \sum_{\alpha \in \Sigma, i \leq n} \tau(\alpha, s_i) M_{(\alpha,i)}$$

Now, given a one-step strategy τ of an MDP or a PFA over S, Σ, applying τ at Δ_1 means going from distribution Δ_1 to distribution $\Delta_2 = \Delta_1 \cdot M_\tau$. A general strategy σ is just an infinite sequence of one-step strategies. Given an MDP or a PFA, an initial distribution Δ and a strategy $\sigma = \tau_1 \ldots$, we define for every $m \in \mathbb{N}$, the (probability) distribution Δ_m^σ over the set of states S reached after m-steps as $\Delta_m^\sigma = \Delta \cdot M_{\tau_1} \cdots M_{\tau_m}$.

2.1 Safety w.r.t. a polytope

Let $\mathcal{A}$ be an MDP or a PFA over n states and let H be a convex polytope in $\mathbb{R}^n$. In most of the paper, we will consider that convex polytopes are defined using the so-called H-representation, that is as an intersection of a finite number of half spaces in $\mathbb{R}^n$, where each half-space or boundary can be written as a linear inequality. Thus, we assume that H is given by a set of inequalities, and denote by $|H|$ the size of this set of inequalities. In section 7, we will consider the V-representation of H, that is the representation given as its finite set of extremal vertices. In this paper, each polytope will be convex and closed (unless explicitly stated otherwise), and we will abusively call them polytopes. Also, all polytopes will be stochastic, that is intersected with half-spaces $\sum_i^n x_i \geq 1$ and $\sum_i^n x_i \leq 1$ to ensure that $\sum_i^n x_i = 1$, and $0 \leq x_i \leq 1$ for all $i \leq n$.

A strategy $\sigma = \tau_1 \ldots$ is said to be *H-safe* from $\Delta_1 \in H$ if for all $m \in \mathbb{N}$, $\Delta_m^\sigma = \Delta_1 \cdot M_{\tau_1} \cdots M_{\tau_m} \in H$. That is, σ is a strategy of $\mathcal{A}$ that allows us to stay forever in H when starting from Δ_1.

Let $H_{\text{win}}^{\mathcal{A}}$ be the set of distributions Δ of H such that there exists a H-safe strategy from Δ, i.e., a strategy σ of $\mathcal{A}$ staying forever in H from Δ. Also, we just write H_{win} when $\mathcal{A}$ is clear from context.

Lemma 2.4. *H_{win} is exactly the set of distributions Δ of H such that there is a one step strategy τ such that $\Delta \cdot M_\tau \in H_{win}$.*

We now state a classical result for MDPs as transformers of probability distributions, which will imply that $H_{\text{win}}^{\mathcal{A}}$ is a convex set for every MDP $\mathcal{A}$. This can also be found in [20, Lemma 2.5], where the result is stated in terms of properties of so-called row-independent Markov set-chains (of which MDPs are an example).

Lemma 2.5. *Let $x, y \in H$ be such that there exist two one-step MDP-strategies τ_x, τ_y with $x \cdot M_{\tau_x} \in H$ and $y \cdot M_{\tau_y} \in H$. Then for every distribution $z \in [x, y]$ (that is $z = \lambda x + (1-\lambda)y, \lambda \in [0,1]$) there is also a one-step MDP-strategy leading from z to H.*

Lemma 2.5 can be trivially extended by induction for the case where one-step strategies τ_x, τ_y are replaced by H-safe (full) strategies σ_x, σ_y, i.e., strategies staying in H forever from x and y:

Lemma 2.6. *Let $x, y \in H$ be such that there exist two H-safe MDP-strategies σ_x, σ_y from x and y. Then for every distribution $z \in [x, y]$ (that is $z = \lambda x + (1-\lambda)y, \lambda \in [0,1]$) there is also a H-safe MDP-strategy σ_z from z.*

Lemma 2.6 implies the convexity of the set $H_{\text{win}}^{\mathcal{A}}$ for $\mathcal{A}$ an MDP:

Proposition 2.7. *Let $\mathcal{A}$ be an MDP. Let $\Delta_1, \ldots, \Delta_k$ be distributions in $H_{win}^{\mathcal{A}}$, and let $\lambda_1, \cdots, \lambda_k \in [0,1]$ such that $\sum_i \lambda_i = 1$. Then $\Delta = \sum_i \lambda_i \Delta_i \in H_{win}^{\mathcal{A}}$.*

$$\left(M_\alpha = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad M_\beta = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \right)$$

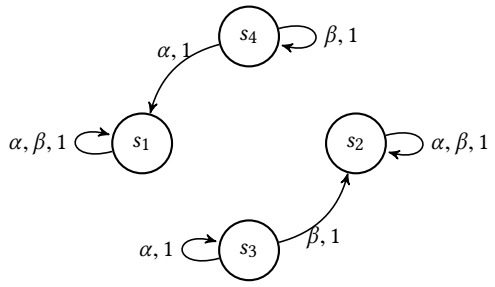

Figure 2. PFA $\mathcal{A}_1$ with two actions α, β

Finally, notice that Lemma 2.5 is not true for PFAs. Indeed, consider a PFA $\mathcal{A}_1$ over 4 states (s_1, s_2, s_3, s_4) as shown in Figure 2. Action α sends the mass from s_4 to s_1, and the remaining mass is kept where they are. Action β sends the mass from s_3 to s_2, and the remaining is kept where they are. Let $H = [\vec{s_3}, \vec{s_4}]$ be the segment from $\vec{s_3}$ to $\vec{s_4}$, that is defined by the half planes $P(s_1) = 0$ (two half planes, one with $P(s_1) \geq 0$ and one with $P(s_1) \leq 0$), $P(s_2) = 0$, $0 \leq P(s_3) \leq 1$, $0 \leq P(s_4) \leq 1$ and $P(s_3) + P(s_4) = 1$.

Consider distributions $x = \vec{s_3}$ and $y = \vec{s_4}$, i.e., $x(s_3) = 1, x(s_1) = x(s_2) = x(s_4) = 0$. Consider the one-step PFA strategies τ_x playing α and τ_y playing β, i.e., $\tau_x(\alpha) = 1$, $\tau_x(\beta) = 0$ and $\tau_y(\beta) = 1, \tau_y(\alpha) = 0$. We have $x \cdot M_{\tau_x} = x \in H$ and $y \cdot M_{\tau_y} = y \in H$. For any $\lambda \in (0, 1)$, consider $z = \lambda x + (1 - \lambda)y$. As the mass in both s_3, s_4 are strictly positive, every one-step strategy τ puts some non-zero mass in s_1 or s_2, and thus goes out of H. Using MDP strategies which can depend upon states, it suffices to play α from s_3 and β from s_4 to have $z \cdot M_\tau = z \in H$, i.e., $\tau(s_3, \alpha) = 1 = \tau(s_4, \beta)$ and $\tau(s_4, \alpha) = 0 = \tau(s_3, \beta)$.

2.2 The problem definitions

In this paper, our focus is on *safety* properties stated on the distributions. We now define the problems we tackle formally.

Definition 2.8 (The existential and universal safety problems for MDPs and PFAs)**.** Given an MDP or a PFA $\mathcal{A}$ over n states, and a closed convex polytope H in $\mathbb{R}^n$,

- the *existential safety problem* asks whether there exists an initial distribution Δ in H and a H-safe strategy of $\mathcal{A}$ from Δ. In other words, is $H^{\mathcal{A}}_{\text{win}} \neq \emptyset$?
- the *universal safety problem* asks whether, for all initial distributions Δ in H, there exists a H-safe strategy of $\mathcal{A}$ from Δ, i.e., is it the case that $H = H^{\mathcal{A}}_{win}$.

The rest of this paper is devoted to solving these problems. We tackle the decidability of these problems, as well as study their complexity, providing both upper and lower bounds.

3 Existential safety for MDPs

In this section, we address the existential safety problem for MDPs and show its decidability.

Theorem 3.1. *The existential safety problem for MDPs is PTIME-complete.*

To understand the difficulty of the question, note that even if we guess a correct Δ, σ, verifying that σ is a H-safe strategy from Δ is highly non-trivial. Indeed, we would need to check for all $m \in \mathbb{N}$, $\Delta^\sigma_m \in H$. As mentioned in the introduction, even in the simple case where there is a single action ($|\Sigma| = 1$), $\mathcal{A}$ is just a Markov chain, and the problem is already as hard as the so-called Skolem problem [3] whose decidability has been opened for decades.

However, when we ask for existence of a safe initial starting distribution, we prove that the problem becomes surprisingly simpler. The main crux of the idea is to prove a fixed point characterization: a H-safe strategy exists iff there exists a strategy that fixes some distribution of H. Thus it suffices to search for (Δ, τ) such that $\Delta = \Delta \cdot M_\tau \in H$. We show that it can be done in polynomial time, by cleverly writing it as a linear program.

For the case where $|\Sigma| = 1$, i.e., there is a single action, one can adapt Tiwari's proof [28] and show that such a fixed point characterization does hold by appealing to Brouwer's fixed point theorem. We cannot lift this directly to the case of MDPs or PFAs since we have multiple actions/matrices. Our main contribution in this section is to show that we can overcome this by exploiting the nice structure of MDPs and obtain a fixed point characterization, by appealing to the more powerful Kakutani's fixed point theorem. To do so, we crucially use the convexity of $H^{\mathcal{A}}_{\text{win}}$, that we proved for an MDP $\mathcal{A}$ in the previous section (essentially inspired from Markov set chain theory [20]). Let us start by recalling the statement of Kakutani's fixed point theorem [21].

Theorem 3.2 (Kakutani's Fixed Point Theorem)**.** *Let S be a non-empty, compact and convex subset of some Euclidean space $\mathbb{R}^n$. Let $f : S \to 2^S$ be an upper hemicontinuous set-valued function on S with the property that $f(x)$ is non-empty, closed and convex for all $x \in S$. Then f has a* fixed point*, i.e., there exists $x \in S$ s.t. $x \in f(x)$.*

Recall that upper-hemicontinuity means that for all open sets O, if $f(a) \subseteq O$, then there is an open set N s.t. $a \in N$ and for all $a' \in N$, $f(a') \subseteq O$. Now, let $\mathcal{A}$ be an MDP. Consider $S = H^{\mathcal{A}}_{\text{win}}$. It is a convex region by Proposition 2.7. It is also closed as H is closed. It is bounded as it is a subset of the set of distributions over n variables, and thus compact as the dimension n is finite. Consider the following function:

Lemma 3.3. *Let $f : H_{win} \to 2^{H_{win}}$ with $f(\Delta) = \{\Delta' \in H_{win} \mid \Delta' = \Delta \cdot M_\tau$ for some one-step strategy $\tau\}$. Then for all $\Delta \in H_{win}$, $f(\Delta) \neq \emptyset$, and f is upper hemicontinuous.*

Proof. The first statement follows directly from Lemma 2.4. For the second statement, assume by contradiction that f is not upper hemicontinuous. Then there is an open set O and $f(a) \subseteq O$, and a sequence a_i converging towards a such that there is $b_i \in f(a_i)$ and $b_i \in (H_{\text{win}} \setminus O)$. As H_{win} is a compact set, we can extract a converging subsequence. Let b be the limit of this sequence. We have that $b \in (H_{\text{win}} \setminus O)$ as $(H_{\text{win}} \setminus O)$ is closed.

Now, by definition of f, we have one step strategies τ_i s.t. $b_i = a_i \cdot M_{\tau_i}$. The space of one-step strategies is trivially compact. So we can again extract from (τ_i) a converging subsequence. Let τ be the limit of this subsequence. Now, $a_i \cdot M_{\tau_i}$ tends towards $a \cdot M_\tau$ by continuity of linear operators. As $a_i \cdot M_{\tau_i} = b_i$, it also converges towards b. Hence $b = a \cdot M_\tau$ (the limit is unique). Thus $b \in f(a) \subseteq O$, that is, $b \in O$, a contradiction with $b \in H_{\text{win}} \setminus O$. □

We now define $X = \{\Delta \in H \mid \Delta = \Delta \cdot M_\tau$ for some one-step strategy $\tau\}$. This is a subset of H_{win}. Using Kakutani's fixed point theorem, we obtain:

Lemma 3.4. *$H_{win} \neq \emptyset$ iff $X \neq \emptyset$.*

Proof. $X \subseteq H_{\text{win}}$, so if $X \neq \emptyset$, then $H_{\text{win}} \neq \emptyset$. If $H_{\text{win}} \neq \emptyset$, by Kakutani's fixed point theorem, there exists a $\Delta \in H_{\text{win}}$ such that $\Delta \in f(\Delta)$ which means that there exists a $\Delta \in H$ and a one-step strategy τ with $\Delta \cdot M_\tau = \Delta$. Hence $\Delta \in X$ and $X \neq \emptyset$. □

One can adapt the proof of Lemma 2.5 to obtain:

Lemma 3.5. *X is a convex set.*

For $i \leq n$, let s_i be a state of the given MDP. We define the weighted outcome of the one-step strategy from s_i to be the set $Im_i = \{\lambda \vec{s_i} \cdot M_\tau \mid \lambda \in [0,1]$, and τ is a one-step strategy$\}$. Let $i \leq n$ and let $\Sigma = \{\alpha_1, \ldots, \alpha_k\}$. Further, for all $j \leq k$, let t_i^j be the distributions obtained as $\vec{s_i} \cdot M_{(\alpha_j, i)}$. For all i, Im_i is a convex set, and more precisely a bounded cone from the origin ($\vec{0} \cdot \vec{s_i}$ for any i) to $(t_i^j)_{j \leq k}$. We have the following lemma:

Lemma 3.6. *Let δ be a sub-distribution. Then, we have $\delta \in Im_i$ iff $\exists \mu^1, \ldots, \mu^k \in [0,1]$ with $\sum_j \mu^j \leq 1$ and $\delta = \sum_j \mu^j t_i^j$.*

Using this Lemma, we obtain the following characterization:

Lemma 3.7. *We have $X \neq \emptyset$, i.e., $\exists \lambda_1, \ldots, \lambda_n \in [0,1]$ such that:*

- *$\Delta = \sum_i \lambda_i \vec{s_i} \in H$ and*
- *there exists a one-step strategy τ with $\Delta \cdot M_\tau = \Delta$.*

iff $\exists \lambda_1, \ldots, \lambda_n \in [0,1]$ and $\exists \mu_1^1, \ldots, \mu_n^k \in [0,1]$, where $k = |\Sigma|$, such that:

(1) $\sum_i \lambda_i \vec{s_i} \in H$ (i.e., it satisfies the linear number of equations associated with H),
(2) For all i, we have $\sum_j \mu_i^j = \lambda_i$,
(3) $\sum_{i,j} \mu_i^j t_i^j = \sum_i \lambda_i \vec{s_i}$.

Now, the second condition in Lemma 3.7 is clearly a set of linear (in)equalities and can be solved using linear programming in polynomial time. As a result we can check if $X \neq \emptyset$ in PTIME. By Lemma 3.4, we conclude that we can check if $H_{\text{win}} \neq \emptyset$ in PTIME.

To complete the proof of Theorem 3.1, it remains to show that this problem, i.e., existential safety for MDPs is indeed PTIME-hard. In fact, it turns out that this is already true for MDPs with $|\Sigma| = 1$, where we take the single matrix M_α to be the identity matrix of dimension n. In this case, the existential safety problem reduces to checking if the convex closed polytope H is empty or not. Given a set of linear inequalities, which is how H is represented to us, checking whether the set of solutions is empty is PTIME-hard (see e.g., [18, Section A.4]). Hence we conclude that existential safety for MDPs is PTIME-complete. This concludes the proof of Theorem 3.1.

4 Existential safety for PFAs

We now turn to the existential safety problem for PFAs. We will show that unlike for MDPs, this problem is undecidable with a mild relaxation on H. Notice that we cannot use the usual undecidability proof for reachability in PFAs, as reachability corresponds to *initialized* safety (given a distribution Δ, is there a H-safe strategy from Δ?). The previous section showed that existential safety for MDPs is much simpler than initialized safety (PTIME instead of being Skolem-hard, even in the unary case where there is a single action [3]), so one might have expected an improvement for PFAs as well.

We show that this is not the case. Inspired by [9], we perform a reduction from the *universal* halting problem for 2-counter machines, which is undecidable (and even Π_2^0-complete), granted that two dimensions of the convex polytope H can be open rather than closed.

Theorem 4.1. *The existential safety problem for PFA is undecidable for convex polytopes having open and closed boundaries.*

The rest of this section will be devoted to the proof of the above theorem. Let CM be a 2-counter machine, with two counters c, d. We want to know whether CM terminates on all inputs. Let pc the program counter, with possible values $\{1, \ldots, n\}$ which is either an increment operation on a counter or a combined zero-test and decrement operation of the form: if $c = 0$ then go to s, else decrement c and go to t.

We will define a PFA $\mathcal{A}$ and a polytope H, such that CM halts for all inputs iff the existential safety is not true, i.e., there exists no $\Delta \in H$ such that there is a H-safe strategy for $\mathcal{A}$ from Δ. The main idea is to encode a counter value as the probability mass in a specific state. Then, when the counter is incremented (or decremented), a "correct" choice of actions will result in the probability mass in that state changing appropriately to encode the incremented (or decremented) counter value. If this correct choice of actions is not taken, then we ensure that the resulting distribution must go outside H and hence is not H-safe. Thus, for any terminating computation of CM, no (correct or faulty) simulation of $\mathcal{A}$ will be H-safe. On the other hand, a non-terminating computation of CM from some initial state will result in a H-safe strategy from a corresponding initial distribution iff the simulation is correct. Formally, we have:

States of the PFA

- *(counter value states)* We have two states C, D encoding the two counters c, d respectively. The counter value $c = j \geq 0$ (resp. for d) will be encoded as a probability mass of $\frac{1}{1000 \cdot 2^j}$ being in C (resp. D). We take this value to be very small, since we want to be able to encode increment and decrement of these states using actions, and for this we need to transfer probability mass from other states. Hence we want this to be small enough to be ensured that there will be some other state (in particular the state T below, from which this probability can be transfered).
- *(program counter state)* The state P will encode the program counter, with $pc = i$ for $1 \leq i \leq n$ being encoded as probability mass of $\frac{i}{1000n}$ in P (values that are not a valid encoding will immediately lead $\mathcal{A}$ out of H),
- *(special states)* S, T are two special states. S is a *stable* state, which will always have probability mass $\frac{1}{10}$ in it and T is a *trash* state which collects all the remaining probability,
- *(verification states)* These states are used to ensure that the above states behave as they should, i.e., the probability mass in them is as specified. More precisely, we have:
  - For each $1 \leq i \leq n$, we have CP_i, CQ_i to check the program counter P encodes $pc = i$.
  - CA, CB, CX, CY, CZ (and similarly DA, DB, DX, DY, DZ) to check that the zero test evaluates to true or false for C (resp. D),

– XC, XD to check that the new value of C and D are as expected.

Defining the polytope H We design the polytope H by specifying $\Delta \in H$ iff the following hold:

(h1) $\Delta(S) = \frac{1}{10}$ (probability mass at S is exactly $\frac{1}{10}$)
(h2) $\Delta(C), \Delta(D) \in (0, \frac{1}{1000}]$ and $\Delta(P), \Delta(CA), \Delta(DA) \in [0, \frac{1}{1000}]$,
(h3) $\sum_{i=1}^{n} \Delta(CQ_i) = \frac{1}{100000n}$,
(h4) $\Delta(CP_i) = \Delta(CQ_i)$ for all i,
(h5) $\Delta(CY) \leq \Delta(CA)$ and $\Delta(CB) = \Delta(CZ)$, and similarly for DA, DB, DY, DZ,
(h6) $\Delta(XC) = \Delta(CX) + \Delta(CY) + \Delta(CZ) \in [0, \frac{1}{2000}]$ and similarly for XD.

Note indeed that the above can be defined as an intersection of half-spaces, using inequalities and further, the space defined is convex.

Actions and Transitions of the PFA From a distribution $\Delta \in H$, assume that there exists a one-step strategy τ such that $\Delta_2 = \Delta \cdot M_\tau \in H$. We will make sure that there is at most one such τ. Recall that $\tau(\alpha)$ represents the proportion of action α which will be played by the strategy (from every state of the PFAs). We will call this weight of action α. Further, in what follows, we say an action α sends p of the mass of state s to state s', to mean that from state s there is a transition labeled α to s' with probability p. When probability p is 1, we just say that the action sends the mass of state s to s'.

$\mathcal{A}$ has (at most) $2n + 4$ actions:

- *Action* ι sends the mass of every state to state T. It will be used to make the sum of weights of actions add up to 1. (That is, from each state, there is a transition labeled ι to T, with probability 1.)
- *Action* δ sends the mass of every state to state T, except for T which is fully sent to S. It will be used to replenish the stable state S (to ensure it has a probability mass of $\frac{1}{10}$ after every step),
- *Action* δ_C sends the mass of every state to T except for S, for which it sends $\frac{1}{40}$ of the mass to XC, $\frac{1}{2}$ to C and the rest to T. Action δ_D is similar, replacing C, XC by D, XD. They will ensure that the probability mass in C, D encode correct counter values.
- There are at most 2 actions α_i, β_i per program counter $pc = i$: one action α_i for increment and two actions α_i, β_i for decrement/zero test. We detail the action α_i encoding the instruction, $pc = i : c \geq 1$, decrement c and goto j:
  1. Send $\frac{1}{10i}$ of the mass of P into CP_i, and the rest into T,
  2. Send all the mass of C into CY,
  3. Send $\frac{1}{2}$ of the mass of D into DX, and the rest to T,
  4. Send $\frac{1}{1000n}$ of the mass of S into CQ_i, $\frac{1}{200}$ of the mass of S into CA, and send $\frac{j}{10n}$ of the mass of S into P, and the rest into T,
  5. Send all the mass of the rest into T.

  This is the only action with β_i which sends mass to CP_i, CQ_i. Assuming $\Delta(P) = \frac{i}{1000n}$ ($pc = i$), because of ($h4$), 1 and 4, only α_i, β_i can have positive weight, because we have for all j, $\Delta_2(CP_j) = \Delta(P)\frac{\tau(\alpha_i)+\tau(\beta_i)}{10j \cdot n} = \Delta_2(CQ_j) = \frac{\tau(\alpha_i)+\tau(\beta_i)}{10000 \cdot n}$, that is $\Delta(P) = \frac{j}{1000n}$ for $\tau(\alpha_j) + \tau(\beta_j) \neq 0$. That is, $\tau(\beta_j) = \tau(\alpha_j) = 0$ for all $j \neq i$. Further, $\tau(\alpha_i) + \tau(\beta_i) = \frac{1}{10}$ thanks to Condition (h3).

  Assuming that $\Delta(C) \leq \frac{1}{2000}$ ($c \geq 1$), because β_i sends $\frac{1}{1000}$ into CB and $\Delta(C)$ into CZ, we must have $\tau(\beta_i) = 0$ to ensure (h5) $\Delta_2(CB) = \Delta_2(CZ)$.

  Thus $\tau(\alpha_i) = \frac{1}{10}$. Further, $\Delta_2(CY) = \frac{\Delta(C)}{10}$ through $\tau(\alpha_i) = \frac{1}{10}$. By Condition (h6), the same mass must enter in XC as $\Delta_2(CX) = \Delta_2(CZ) = 0$. Hence $\tau(\delta_c)/400 = \Delta(C)/10$ which means $\tau(\delta_c) = 40\Delta(C)$. So the mass entering C through $\tau(\delta_C)$ is $40\Delta(C) * 1/20 = 2\Delta(C)$ which is equivalent to c being decremented. In the same way, we can observe that the mass in counter d remains unchanged through δ_D.
- *Action* β_i *coding* $pc = i : c = 0$ *and goto* j is as follows:
  1. Send $\frac{1}{10i}$ of the mass of P into CP_i, and the rest into T,
  2. Send $\frac{1}{2}$ of the mass of C into CZ, and the rest into T,
  3. Send $\frac{1}{2}$ of the mass of D into DX, and the rest to T,
  4. Send $\frac{1}{1000n}$ of the mass of S into CQ_i, $\frac{1}{200}$ of the mass of S into CB, and send $\frac{j}{10n}$ of the mass of S into P, and the rest into T,
  5. Send all the mass of the rest into T.

  As above, we have $\tau(\alpha_i) + \tau(\beta_i) = \frac{1}{10}$. Assuming that $\Delta(C) = \frac{1}{1000}$ ($c = 0$), because α_i sends $\frac{\tau(\alpha_i)}{2000}$ into CA and $\tau(\alpha_i) \cdot \Delta(C) = \frac{\tau(\alpha_i)}{1000}$ into CY, we must have $\tau(\alpha_i) = 0$ to ensure (h5) $\Delta_2(CY) = \Delta_2(CA)$.

  Hence $\tau(\beta_i) = \frac{1}{10}$. Thus, $\Delta(C)/20$ enters CZ. By Condition (h6), the same mass must enter in XC as $\Delta_2(CX) = \Delta_2(CY) = 0$. Hence $\tau(\delta_c)/400 = \Delta(C)/20$ which means $\tau(\delta_c) = 20\Delta(C)$. So the mass entering C through $\tau(\delta_C)$ is $20\Delta(C) * 1/20 = \Delta(C)$ which is equivalent to c staying at $\frac{1}{1000}$, that is the counter c stays at $c = 0$. In the same way, we can observe that the mass in counter d remains unchanged through δ_D.
- *Action* α_i *encoding* $pc = i$: *increment* c *and goto* j is as follows:
  1. Send $\frac{1}{10i}$ of the mass of P into CP_i, and the rest into T,
  2. Send $\frac{1}{4}$ of the mass of C into CX, and the rest into T,
  3. Send $\frac{1}{2}$ of the mass of D into DX, and the rest to T,
  4. Send $\frac{1}{1000n}$ of the mass of S into CQ_i, and send $\frac{j}{10n}$ of the mass of S into P, and the rest into T,
  5. Send all the mass of the rest into T.

  This is the only action (β_i does not exists as this is an increment) which sends mass to CP_i, CQ_i. Assuming $\Delta(P) = \frac{i}{1000n}$ ($pc = i$), because of ($h4$), 1 and 4, only this action can have positive weight, that is $\tau(\beta_j) = \tau(\alpha_j) = 0$ for all $j \neq i$. Further, $\tau(\alpha_i) = \frac{1}{10}$ thanks to Condition (h3).

  Further, $\Delta(C)/40$ enters CX through $\tau(\alpha_i)$. By Condition (h6), the same mass must enter in XC as $\Delta_2(CY) = \Delta_2(CZ) = 0$. Hence $\tau(\delta_c)/400 = \Delta(C)/40$ which means $\tau(\delta_c) = 10\Delta(C)$. So the mass entering C through $\tau(\delta_C)$ is $10\Delta(C) * 1/20 = \Delta(C)/2$ which is equivalent of c being incremented. In the same way, we can observe that the mass in counter d remains unchanged through δ_D.

We obtain a correct simulation from distributions corresponding to configurations of the 2-counter machine. In particular, there exists a safe strategy from this distribution iff the computation from the corresponding configuration is not halting. We obtain that the PFA is existentially safe iff M is not universally hatling.

Notice that ($h2$) has some strict inequalities, asking $\Delta(C), \Delta(D) > 0$. This is to avoid considering configurations with infinite counters, from which there may exist a non-halting computation.

5 Universal safety for MDPs

In this section, we prove that universal safety is decidable for MDPs. Further, we provide tight complexity bounds:

Theorem 5.1. *The universal safety problem for MDPs is co-NP-complete.*

Our first step is to express universal safety as a property on the one-step strategies.

Lemma 5.2. *Let M be an MDP and H a convex polytope. Then $H = H_{win}$ iff for any distribution Δ in H, there exists a one-step strategy τ (of the MDP) which sends in H, that is $\Delta \cdot M_\tau \in H$.*

Proof. If for each distribution $\Delta \in H$, there exists such a one-step strategy τ_Δ, then one can extend it to a distribution-based strategy playing τ_Δ when in Δ. That is, for each $\Delta \in H$, it suffices to play the strategy σ defined inductively by $\sigma(1) = \tau_\Delta$ and $\sigma(n+1) = \tau_{\Delta_n}$ with $\Delta_n = \Delta \cdot M_{\sigma(1)} \cdots M_{\sigma(n)}$. We prove trivially by induction that $\Delta_n \in H$, and thus τ_{Δ_n} is well defined and $\Delta_{n+1} \in H$. Thus, σ is a H-safe strategy from Δ. Thus $H \subseteq H_{\text{win}}$. But by definition we know that $H_{\text{win}} \subseteq H$, which implies that $H = H_{\text{win}}$.

Conversely, if $H = H_{\text{win}}$, then for all $\Delta \in H$ we have $\Delta \in H_{win}$. Thus there is a strategy staying forever in H from any $\Delta \in H$, and in particular a one-step strategy staying in H. □

5.1 A co-NP upper bound for universal safety in MDPs

Our goal is to check the characterization in Lemma 5.2 by encoding it as a *quantified* linear program and exploiting advances and the state-of-the-art results in the theory of linear arithmetic and linear inequalities [15, 31]. For this we first obtain another intermediate characterization, which brings us closer to our goal. We reuse the notation (t_i^j) of Section 3, defined as the distributions $\vec{s_i} \cdot M_{(\alpha_j, i)}$.

Lemma 5.3. *Let $\mathcal{A}$ be an MDP, with set of states S and actions Σ, where $k = |\Sigma|$, $n = |S|$. Let H be a convex set. Then the following are equivalent:*

- *(P1) $H = H_{win}$*
- *(P2) for all distributions $\Delta \in H$, there exists a one-step strategy τ such that $\Delta \in H$ implies that $\Delta \cdot M_\tau \in H$*
- *(P3) for all $\lambda_1, \ldots \lambda_n \in [0,1]$, there exists $\mu_1^1, \ldots, \mu_n^k \in [0,1]$, such that $\sum_i \lambda_i \vec{s_i} \in H$ (it satisfies the linear number of inequalities associated with H) implies that:*
  - *a. For all i, we have $\sum_j \mu_i^j = \lambda_i$,*
  - *b. $\sum_{i,j} \mu_i^j t_i^j \in H$ (it satisfies the linear number of inequalities associated with H).*

Proof. The statement (P1) iff (P2) follows from Lemma 5.2.

Now we prove (P2) iff (P3). Recall that Im_i (see Section 3) is the weighted outcome of one-step strategy from $\vec{s_i}$, denoted as $Im_i = \{\lambda \vec{s_i} \cdot M_\tau \mid \lambda \in [0,1]$, and τ is a one-step strategy $\}$. The proof follows ideas of Lemmas 3.6, 3.7. Assume (P2). Let $(\lambda_i)_{i \leq n}$ such that $\Delta = \sum_i \lambda_i \vec{s_i} \in H$. Thus there exists a τ with $\Delta \cdot M_\tau \in H$. Let $\nu_i^j = \tau(\alpha_j, \vec{s_i})$. We have $\Delta \cdot M_\tau = \sum_{i,j} \lambda_i \nu_i^j t_i^j \in H$. For all i, j, choosing $\mu_i^j = \lambda_i \nu_i^j$ satisfies a and b. Hence (P3) is true.

Assume (P3). Let $\Delta = \sum_i \lambda_i \vec{s_i} \in H$. It suffices to consider τ such that $\tau(\alpha_j, \vec{s_i}) = \frac{\mu_i^j}{\lambda_i}$ for $\lambda_i > 0$ and $\tau(\alpha_j, \vec{s_i}) = 0$ otherwise to prove (P2). □

Now, we observe that ($P3$) is a *quantified linear implication (QLI)*, i.e., a conjunction of implications of inequalities over real numbers of the form:

$$\exists \mathbf{x_1} \forall \mathbf{y_1} \ldots \exists \mathbf{x_n} \forall \mathbf{y_n} [\mathbf{A} \cdot \mathbf{x} + \mathbf{N} \cdot \mathbf{y} \leq \mathbf{b} \rightarrow \mathbf{C} \cdot \mathbf{x} + \mathbf{M} \cdot \mathbf{y} \leq \mathbf{d}]$$

where, $\mathbf{A}, \mathbf{N}, \mathbf{C}, \mathbf{M}$ are matrices and $\mathbf{x}, \mathbf{y}, \mathbf{b}, \mathbf{d}$ are vectors partitioned respectively as $x_1, \ldots x_n$ and $y_1, \ldots, y_n$. The decidability of solving (checking existence of a solution for) such QLI's with an arbitrary quantifier alternation is known to be PSPACE-hard [15]. But it turns out that our specific problem has a better structure which allows us to use recently proved results in [31] and show the following:

Proposition 5.4. *Solving the quantified linear implication ($P3$) can be done in co-NP.*

Proof. First, we observe that ($P3$) has a single alternation between universally quantified variables and existentially quantified variables, further, the first variable is universally quantified. In the notation of [15, 31], this means that the problem ($P3$) is in the class denoted by QLI(1,∀,**B**)[1]. This allows us to appeal to Theorem 6 of [31] (or see Lemma 5.1 of [30] for an alternate proof) that states that this class QLI(1,∀,**B**) is co-NP-complete. Thus, we obtain that ($P3$) is in co-NP. □

Since solvability for this class of QLI is co-NP-hard as well [15], one may try to prove that these particular instances are actually as hard as general QLP(1,∀,**B**) questions. The difficulty is that the equations on the right hand side and on the left hand side are both the same equations associated with H, which is a very special case of the general QLI(1,∀,**B**) class and it is not immediately clear how to transform an arbitrary QLI from this class to an instance of ($P3$). Nevertheless, we next show a direct proof of co-NP-hardness.

5.2 A co-NP lower bound for universal safety in MDPs

We now prove a matching lower bound, showing that we cannot hope to find a PTIME algorithm for universal safety in general MDPs (unless PTIME = NP):

Proposition 5.5. *Checking universal safety for MDPs is co-NP-hard.*

The proof is by a reduction from the complement of 3-CNFSAT, which is co-NP-complete. The complement asks, given a 3-CNFSAT formula, if it is uniformly false, i.e., whether for all valuations, there exists a clause which evaluates to false.

Let $x_1, \ldots x_n$ be the variables and $c_1, \ldots, c_k$ be the clauses (in 3-CNF) of the formula Φ. We let $m = max(k, n)$, be the maximum between the number of variables and the number of clauses.

Our goal is to define an MDP and a polytope H such that H is universally safe iff Φ is not satisfiable. By the characterization in Lemma 5.2, H is universally safe iff from any initial distribution in H, there exists a one-step strategy of $\mathcal{A}$ that remains in H. Thus we will in fact design an MDP $\mathcal{A}$ and a polytope H such that from any initial distribution in H, there exists a one-step strategy τ of $\mathcal{A}$ that remains in H iff Φ is not satisfiable.

[1] **B** refers to the fact that both existentially/universally quantified variables may occur in both sides of the implication. In fact, we fall in a restriction where existentially quantified variables only occur on Right hand side, but this doesn't change the complexity.

The states of the MDP will correspond to the variables and clauses, as defined later. We start by defining the alphabet of actions for the MDP, of size $2nk + 2$:

- for each $1 \le i \le n$ and $1 \le j \le k$, we will have two actions α_i^j, β_i^j that are associated with variable x_i and clause c_j,
- one action δ to replenish a "stable" state and one action ι to ensure that the weight of outgoing actions sums up to 1.

We also introduce a notation. For any clause c_j, we denote γ_1^j for α_i^j if x_i is the first literal of c_j, and γ_1^j for β_i^j if $\neg x_i$ is the first literal of c_j, and similarly for the second and third literals of c_j.

A high level intuition of the proof Each valuation v will correspond to an initial distribution Δ_v. Given a valuation v for variables $x_1, \ldots, x_n$, we need to check if there is any clause c_j which is false, i.e., such that all literals of c_j are set to false by v. To find such a j, we will let the one-step strategy τ choose *uniformly* the clause c_j which is false: there must be a j such that for all i, either α_i^j has positive weight or β_i^j has positive weights (that is, the sum of the two weights is non zero).

For that, we design H to ensure that if $\Delta_v \cdot M_\tau \in H$, then:

I1 for all i, $\sum_j \tau(\alpha_i^j) + \tau(\beta_i^j) = \frac{1}{20m}$.

I2 for all j and all i, i', $\tau(\alpha_i^j) + \tau(\beta_i^j) = \tau(\alpha_{i'}^j) + \tau(\beta_{i'}^j)$.

I3 for all j, $\tau(\gamma_1^j) = \tau(\gamma_2^j) = \tau(\gamma_3^j) = 0$,

I(v) $\sum_j \tau(\beta_i^j) = 0$ for all i such that x_i is true under v, and $\sum_j \tau(\alpha_i^j) = 0$ for all i such that x_i is false under v.

We first want to show that for a given valuation v, if there is a clause c_j which is false under v, then there is a one step strategy $\tau_{v,j}$ with $\tau_{v,j}$ satisfying the conditions I(v), I1, I2, I3. This strategy is defined as follows:

J1. $\tau_{v,j}(\alpha_i^{j'}) = \tau_{v,j}(\beta_i^{j'}) = 0$ for $j' \neq j$,

J2. $\tau_{v,j}(\alpha_i^j) = \frac{1}{20m}, \tau_{v,j}(\beta_i^j) = 0$, if variable x_i is true under v,

J3. $\tau_{v,j}(\beta_i^j) = \frac{1}{20m}, \tau_{v,j}(\alpha_i^j) = 0$, if variable x_i is false under v,

For v, j such that c_j is false under v, we indeed have that $\tau_{v,j}$ satisfies I(v), I1, I2, I3. First, J1,J2,J3 imply I(v),I1,I2 for all j. For I3, for all $j' \neq j$, J1 implies that $\tau_{v,j}(\gamma_1^{j'}) = \tau_{v,j}(\gamma_2^{j'}) = \tau_{v,j}(\gamma_3^{j'}) = 0$. To show I3 for the remaining case, i.e., when $j' = j$, we remark that as c_j is false under v, we have I3: all literals of c_j are set to false by v, so $I(v)$ (which we already proved) ensures that $\tau_{v,j}(\gamma_1^j) = \tau_{v,j}(\gamma_2^j) = \tau_{v,j}(\gamma_3^j) = 0$. Thus I3 is true.

Conversely, we want to show that with such an H, for all valuations v, if a one-step strategy τ satisfies I(v), I1, I2, I3, then there is a clause c_j which is false under v (there may be several such clauses, and the strategy may choose several of them, as long as it does so uniformly (because of I2) for all i).

Consider such a τ. Now, because of I1, for all i, there is some j_i such that $\tau(\alpha_i^{j_i}) + \tau(\beta_i^{j_i}) > 0$. Because of I2, we know that we can choose j uniform in i, i.e., for all i, $j_i = j$. We can apply I3 for this j, implying that $\tau(\gamma_1^j), \tau(\gamma_2^j), \tau(\gamma_3^j)$ are all null. Using $I(v)$, we have that c_j is false under v. Indeed, assume by contradiction that some literal of c_j is true under v. Wlog, we can assume that it is the first literal of c_j, and that this literal is e.g. $\neg x_i$, i.e., x_i false under v. As $\tau(\alpha_i^{j_i}) + \tau(\beta_i^{j_i}) > 0$, and $\tau(\beta_i^{j_i}) = \tau(\gamma_1^j) = 0$, we have $\tau(\alpha_i^{j_i}) > 0$, which is in contradiction with I(v) and x_i false under v. Thus, there exists a j such that c_j is false under v.

Finally, remark that in the forward direction, we need to define one-step strategies τ from all $\Delta \in H$ (so far, we did it only from $\{\Delta_v \mid v \text{ a valuation}\}$). To do this, we define valuation v such that Δ_v is in some sense (made precise later) close to Δ. We show that if there is a clause c_j false under v, then one can play $\tau_{v,j}$ from Δ and stay in H. Notice that when Φ is true under v, there may be some τ defined from Δ but no τ from Δ_v.

Formal construction

States of the machine We have $nk + 3n + 3k + 2$ states:

- For each variable x_i, we associate 3 states X_i, Y_i, Z_i, which will be used to ensure I1 and I(v),
- For each clause c_j and variable x_i, we associate the state C_i^j which will be used to ensure I2,
- For each clause c_j, we associate the states G_1^j, G_2^j, G_3^j which will be used to ensure I3,
- One "stable" state S (containing $\frac{1}{10}$, ensured by polytope H),
- One "trash" state T, which will get the rest of the probability mass (which will be at least $\frac{1}{2}$).

Polytope The polytope H is defined as follows (as done before, we write constraints, but it is easy to see that these can be captured as intersection of half-spaces, linear inequalities):

(Hi) for all i, $\Delta(Y_i) + \Delta(Z_i) = \frac{1}{400m}$, which is used to ensure $I1$,

(Hii) For all $j \le k$ and all $i \neq i' \le n$, $\Delta(C_i^j) = \Delta(C_{i'}^j)$ which is used to ensure $I2$,

(Hiii) For each $j \le k$, we have $\Delta(G_1^j) = \Delta(G_2^j) = \Delta(G_3^j) = 0$ which is used to ensure $I3$,

(Hiv) $\Delta(S) = \frac{1}{10}$,

(Hv) $\Delta(X_i) \in [0, \frac{1}{10m}]$ for all $i \le n$, which encodes the valuation of x_i,

(Hvi) $\Delta(Y_i) \in [0, \frac{1}{400m}]$ for all $i \le n$, which is associated with the weight of action α_i,

(Hvii) $\Delta(X_i) - 20\Delta(Y_i) \in [0, \frac{1}{20m}]$ for all $i \le n$, which enforces I(v),

Actions: Every action sends all the mass from X_i to X_i, and all the mass from $Y_i, Z_i, C_i^j, G_\ell^j$ to T for all $i \le n, j \le k$ and $\ell \in \{1, 2, 3\}$. All actions except δ send all the mass from T to T. Action δ sends all the mass from T to S.

The main difference in the actions is what happens from the single state S. That is why this lower bound applies to MDPs (and PFAs): choosing actions based on state does not make a difference.

Action ι sends the mass from S to T, while δ sends all the mass from S to S.

Actions α_i^j, β_i^j transform the mass of S as follows:

- $\frac{1}{2}$ into Y_i for α_i^j and $\frac{1}{2}$ into Z_i for β_i^j. This combined with (Hi) implies I1 and combined with (Hvii) implies I(v),
- $\frac{1}{20m}$ into C_i^j for both. This combined with (Hii) implies I2,
- α_i^j (resp. β_i^j) sends $\frac{1}{20m}$ into G_ℓ^j if it is γ_ℓ^j. This combined with (Hiii) implies I3,
- the rest of the mass of S is sent back to S.

Enforcing I(v). Let v be a valuation. We associate to v a distribution $\Delta_v \in H$ such that $\Delta_v(X_i) = 0$ if x_i is false under v, and $\Delta_v(X_i) = \frac{1}{10m}$ if x_i is true under v. The mass in S is $\Delta_v(S) = \frac{1}{10}$ and other states can have arbitrary mass as long as $\Delta_v \in H$ (such $\Delta_v \in H$ exists for every valuation v).

Let τ be a one-step strategy such that $\Delta_2 = \Delta_v \cdot M_\tau \in H$. For all $i \leq n$, we denote by a_i the sum of weights $\sum_j \tau(S, \alpha_i^j)$ from state S of action α_i^j for $j \leq k$. Similarly we denote by $b_i = \sum_j \tau(S, \beta_i^j)$. For all i, we have $\Delta_2(Y_i) = \frac{a_i}{20}$ by construction, as $\Delta_v(S) = \frac{1}{10}$. Also $\Delta_2(X_i) = \Delta_v(X_i)$ because all actions send all mass from X_i to X_i.

Now, assume that $\Delta_v(X_i) = 0$ (i.e., x_i is false under v). Then, we have $\Delta_2(X_i) = 0$ by construction. As $\Delta_2(X_i) - 20\Delta_2(Y_i) \geq 0$ and $\Delta_2(Y_i) \geq 0$, it forces $\Delta_2(Y_i) = 0$ and thus $a_i = 0$.

In the same way, for $\Delta_v(X_i) = \Delta_2(X_i) = \frac{1}{10m}$ (x_i is true under v), we have $\Delta_2(Y_i) \geq \frac{1}{400m}$. Because of (Hi), we have $\Delta_2(Z_i) = 0$, which implies that $b_i = 20m\Delta_2(Z_i) = 0$. That is, I($v$) is ensured.

Notice that once $\tau(S, \alpha_i^j), \tau(S, \beta_i^j)$ have been chosen, there is exactly one choice of weight $\tau(T, \delta)$ of δ which ensures that $S = \frac{1}{10}$, and the rest of the weight of τ from every state goes to ι (T contains at least $\frac{1}{2}$ of the probability mass because the sum of the maximum of all other state is less than half. Also, with the previously defined choice of actions, there is at least $\frac{1}{2}$ of the weight left which can be assigned to δ).

To complete the proof, we sketch that the following statements are equivalent (the formal details can be found in [5]):

(i) H is universally safe
(ii) for all valuations v, there exists a τ such that $\Delta_v \cdot M_\tau \in H$
(iii) the 3CNF formula Φ is uniformly false.

(i) implies (ii) is trivial. Assume (ii). For all valuations v, let τ_v be such that $\Delta_v \cdot M_{\tau_v} \in H$. As sketched in the high-level description, it implies that some clause c_j is false under v, which implies (iii). Finally, assume (iii). Then, consider a distribution $\Delta \in H$. We will associate a valuation v to Δ. For all i, either $\Delta(X_i) \leq \frac{1}{20m}$ and one can choose v setting x_i to false ($\Delta_2(Y_i) = 0$). Otherwise, $\Delta(X_i) = \Delta_2(X_i) > \frac{1}{20m}$ and one can choose v setting x_i to true ($\Delta_2(Y_i) = \frac{1}{400m}$). As (iii) is true, we have some c_j false under v. Applying the one-step strategy $\tau_{v,j}$ sketched in the high-level description yields: $\Delta \cdot M_{\tau_{v,j}} \in H$, which implies that (i) holds.

6 Universal safety for PFAs

Finally, we show that the universal safety problem for PFAs is still decidable, but with a higher complexity of EXPTIME.

Theorem 6.1. *The universal safety problem for PFAs can be solved in EXPTIME and is co-NP-hard.*

Proof. The hardness follows by observing that the proof of co-NP-hardness for MDPs, works *mutatis-mutandis* for PFAs. Hence, universal safety is also co-NP-hard for PFAs.

Next, we observe that universal safety continues to be a property on one-step strategies. In other words, Lemma 5.2 and its proof holds verbatim for PFAs as well. From this, for universal safety of PFAs, it suffices to check the following proposition in the First Order Theory of Reals (denoted Th($\mathbb{R}$)): is it the case that for all $\lambda_1, \ldots, \lambda_n \in [0,1]^n$ with $\sum \lambda_i \vec{s_i} \in H$, there exist $\mu_1, \ldots, \mu_k \in [0,1]^k$ with $\sum_j \mu_j = 1$ and $\sum_{i,j} \lambda_i \mu_j \vec{s_i} \cdot M_{\alpha_j} \in H$. There, $(\lambda_i)_{i \leq n}$ represent the coordinates over the basis $\vec{s_1}, \ldots, \vec{s_n}$ of a distribution $\Delta = \sum_i^n \lambda_i \vec{s_i} \in H$, while $(\mu_j)_{j \leq k}$ are the coefficients of the one-step strategy τ with $\tau(\alpha_j) = \mu_j$ for actions $\alpha_1, \ldots, \alpha_k$.

It is well known that Th($\mathbb{R}$)is in 2EXPTIME, which gives decidability in 2EXPTIME for this problem. Note that since we have PFAs, we cannot exploit the convexity of H_{win} as in MDPs, to encode the problem in quantified variants of linear programming.

In the following, we will show that we can improve this result from 2EXPTIME to EXPTIME. The main idea is that we reduce the above question to an equivalent *existential* FO (denoted $\exists$-Th($\mathbb{R}$)) formula, which involves an exponential blowup.

Consider $t_i^j = \vec{s_i} \cdot M_{\alpha_j}$ obtained from $\vec{s_i}$ playing action α_j. For $\delta = \sum \lambda_i s_i$. Let $\Delta = \sum_i \lambda_i \vec{s_i} \in H$. We can define $Im(\Delta) = \{\sum_{i,j} \lambda_i \mu_j t_i^j \mid \mu_1, \ldots, \mu_k \in [0,1]^k, \sum_j \mu_j = 1\}$. We have $Im(\Delta)$ is convex: given $\Gamma_1, \Gamma_2 \in Im(\Delta)$, associated with $(\mu_j), (\nu_j)$ and given $\ell \in [0,1]$, it suffices to choose $\kappa_j = \ell\mu_j + (1-\ell)\nu_j$ for all j to prove that $\ell\Gamma_1 + (1-\ell)\Gamma_2 \in Im(\Delta)$. Further, $Im(\Delta)$ have k corner points, one for each $j \leq k$, obtained with $\mu_j = 1$, defined as $\sum_i \lambda_i t_j^i$.

Using the separation theorem (consequence of Hahn-Banach theorem), $Im(\Delta) \cap H = \emptyset$ iff there exists an hyperplane K which separates $Im(\delta)$ and H iff there exists K a half space with $K \cap H = \emptyset$ and $Im(\Delta) \subseteq K$.

Thus, we can rewrite the above condition as: Does there exist $\lambda_1, \ldots, \lambda_n \in [0,1]^n$ with $\sum_i \lambda_i s_i \in H$ and a half space K (linear number of equations to existentially guess) disjoint of H (need to check that every corner point is not in K), such that $\sum_i \lambda_i t_i^j \in K$ for all $j \leq k$ (linear number of equations). Notice that for general H under the H-representation, the number of corner points is exponential in $|H|$.

Now, we exploit the fact that there are algorithms for *existential* F0 over reals that run in $O\left(L(md)^{n^2}\right)$[19] where L is the number of bits needed to represent the formula, m is the number of polynomials in the FO sentence, d is the max-degree of polynomials and n is the number of variables. For general H, L and n polynomial in input size, d is a constant and m is exponential in input size. Note that even with m being exponential the run time is still an exponentially bounded function and we obtain an EXPTIME upper bound. □

7 Polytopes under the V-representation.

The above proof for PFAs suggests that the input representation of the polytope is very important. Indeed, the exponential blowup in the above result for PFAs is due to the fact that polynomially many linear equations can define a polytope with exponentially many corner points. This motivates us to consider another representation of convex polytopes, called the V-representation, which gives as input the set $corner(H)$ of r corner points $\Gamma_1, \ldots, \Gamma_r$ of the convex polytope H. With this representation, checking for $\sum_i^n \lambda_i \vec{s_i} \in H$ is done by asking whether there exists $\nu_1, \ldots, \nu_r \in [0,1]^r$ such that $\sum_i^n \lambda_i \vec{s_i} = \sum_j^r \nu_j \Gamma_j$. Existential safety is thus still in PTIME for MDPs, and still undecidable for PFAs.

On the other hand, for universal safety we get better upper bounds, when the polytope is given in the V-representation. For PFAs, it suffices to use the proof of Theorem 6.1, and remark that the number of vertices is polynomial in the input size in this case. We can therefore write this in $\exists$-Th($\mathbb{R}$) whose complexity is in the class $\exists\mathbb{R} \subseteq$ PSPACE (see [27] for a formal definition of this class). For MDPs, we can improve the complexity even further obtaining a PTIME upper bound matching existential safety for MDPs. That is,

Theorem 7.1. *Let H be a polytope given by its V-representation, then solving universal safety is in PTIME for MDPs and $\exists\mathbb{R}$ for PFAs.*

For MDPs, using the convexity of H_{win} (Lemma 2.5), we show that it suffices to test safety from $corner(H)$. For each of the linearly many distributions in $corner(H)$, this can be done in PTIME.

Complexity of safety	MDPs	PFAs
Existential	PTIME	undecidable
Universal	PTIME	$\exists\mathbb{R} \subseteq$PSPACE

Table 2. Complexity for polytopes under the V-representation.

Lemma 7.2. *Let M be an MDP. Then $H = H_{win}$ iff for all distribution Δ in $corner(H)$, there exists a one-step strategy τ with $\Delta \cdot M_\tau \in H$.*

Further, given $\Delta \in H$, checking whether there exists a one-step strategy τ with $\Delta \cdot M_\tau \in H$ can be done in PTIME.

Proof. One direction is trivial. For the other direction, if $corner(H) \subseteq H_{\text{win}}$, then as H_{win} is convex, looking at the convex hull, we obtain $H = hull(corner(H)) \subseteq H_{\text{win}} \subseteq H$ and we get the equality.

For the second statement, let $A = \{\alpha_1, \cdots, \alpha_k\}$ be the actions. Let $(\lambda_i)_{i \le n}$ be coordinates of Δ, i.e, $\Delta = \sum \lambda_i \vec{s_i} \in H$. A one-step strategy τ of an MDP is given by a tuple $(\mu_i^j)_{i \in \{1,\ldots,n\}}^{j \in \{1,\ldots,k\}}$ s.t. for all $i \le n$, the mix of actions $\sum_{j=1}^k \mu_i^j \alpha_j$ is played by τ from state s_i, with $\sum_{j=1}^k \mu_i^j = 1$. For each $\alpha_j \in A$ and each state s_i, we let t_i^j be the distribution reached from s_i playing α. We thus have $\exists \tau$ such that $\Delta \cdot M_\tau \in H$ iff $\exists \nu_1, \ldots, \nu_r, \mu_1^1, \ldots, \mu_n^k \in [0,1]^{r+nk}$ such that $\sum_{i,j} \lambda_i \mu_i^j t_i^j = \sum_i^r \nu_i \Gamma_i$, i.e., a set of linear inequalities (as (λ_i, Γ_i) are given). This is a linear program which can be solved in PTIME. □

8 Conclusion

In this paper, we have defined and analyzed the dynamic behavior of MDPs and PFAs via distribution-based objectives. Our results are summarized in Table 1 (in the Introduction) and Table 2 (above). We obtained tight complexity results for MDPs and safety objectives defined by convex polytope in the usual H-representation, with PTIME-completeness for the existential question and co-NP-completeness for the universal question. When the polytopes are given in the V-representation, we obtain better upper bounds, namely PTIME even for universal safety. These efficient complexity results are surprising, especially in light of the initialized safety problem (i.e., safety from a given initial distribution), which is at least Skolem-hard [3], and is not known to be decidable.

Concerning PFAs, the complexities are higher than MDPs, which is unsurprising. The gap between MDPs and PFAs is large for existential safety (undecidable vs PTIME), while not as much for universal safety (EXPTIME vs co-NP). Interestingly, universal safety has better complexity than existential safety for PFAs, while it is the opposite for MDPs.

We would like to highlight that proving these results required us to use a wide variety of techniques: from (quantified) linear programming to theory of reals, fixed point theorems and SAT/2-counter machine reductions, illustrating the richness of this topic.

In this paper, we considered safety objectives as they are natural and have been considered in simpler deterministic contexts [24, 28]. In terms of futurework, distribution-based objectives are not restricted to safety problems. Another natural question is the escape problem, where we ask for the existence of a strategy escaping the convex polytope H, or equivalently whether all strategies are safe (stay inside H). In deterministic settings (i.e., with a single alphabet), both problems coincide as there is a unique strategy. Further, our decidability results are in a setting where the convex polytope is also closed (while undecidability for PFA required both open and closed boundaries). We believe these can be strengthened (to having open and closed boundaries), but this would require some new techniques. We leave this as well as tackling the non-convex setting for futurework.

References

[1] M. Agrawal, S. Akshay, B. Genest, and P. S. Thiagarajan. 2012. Approximate verification of the symbolic dynamics of Markov chains. In *LICS'12*. IEEE Computer Society, 55–64.

[2] M. Agrawal, S. Akshay, B. Genest, and P. S. Thiagarajan. 2015. Approximate Verification of the Symbolic Dynamics of Markov Chains. *J.ACM* 62(1) (2015), 183–235.

[3] S Akshay, Timos Antonopoulos, Joël Ouaknine, and James Worrell. 2015. Reachability problems for Markov chains. *Inform. Process. Lett.* 115, 2 (2015), 155–158.

[4] S. Akshay, Blaise Genest, Bruno Karelovic, and Nikhil Vyas. 2016. On Regularity of unary Probabilistic Automata.. In *STACS'16*. LIPIcs, 8:1–8:14.

[5] S. Akshay, Blaise Genest, and Nikhil Vyas. 2018. *Distribution-based objectives for Markov Decision Processes.* Technical Report. CoRR, https://arxiv.org/submit/2240184.

[6] D. Beauquier, A. Rabinovich, and A. Slissenko. 2002. A Logic of Probability with Decidable Model Checking. In *CSL'02*. 306–321.

[7] Alberto Bertoni. 1974. The solution of problems relative to probabilistic automata in the frame of the formal languages theory. In *GI Jahrestagung*. 107–112.

[8] Nathalie Bertrand, Miheer Dewaskar, Blaise Genest, and Hugo Gimbert. 2017. Controlling a Population.. In *Concur'17 (LiPIcs)*. 12:1–16.

[9] Vincent D Blondel, Olivier Bournez, Pascal Koiran, Christos H Papadimitriou, and John N Tsitsiklis. 2001. Deciding stability and mortality of piecewise affine dynamical systems. *Theoretical Computer Science* 255, 1 (2001), 687–696.

[10] R. Chadha, Dileep Kini, and M. Viswanathan. 2014. Decidable Problems for Unary PFAs. In *QEST*. LNCS 8657, 329–344.

[11] R. Chadha, V. Korthikanti, M. Vishwanathan, G. Agha, and Y. Kwon. 2011. Model checking MDPs with a Unique Compact Invariant Set of Distributions. In *QEST'11*. 121–130.

[12] K. Chatterjee and M. Tracol. 2012. Decidable Problems for Probabilistic Automata on Infinite Words. In *LICS*. IEEE Computer Society, 185–194.

[13] Laurent Doyen, Thierry Massart, and Mahsa Shirmohammadi. 2012. *Infinite Synchronizing Words for Probabilistic Automata (Erratum).* Technical Report. CoRR abs/1206.0995.

[14] Laurent Doyen, Thierry Massart, and Mahsa Shirmohammadi. 2014. Limit Synchronization in Markov Decision Processes. In *Proceedings of FoSSaCS'14 (Lecture Notes in Computer Science)*, Vol. 8412. Springer, 58–72.

[15] P Eirinakis, S Ruggieri, K Subramani, and P Wojciechowski. 2014. On quantified linear implications. *Ann. Math. Artif. Intell.* 71, 4 (2014), 301–325.

[16] N. Fijalkow, H. Gimbert, and Y. Ouahladj. 2012. Deciding the Value 1 Problem for Probabilistic Leaktight Automata. In *LICS*. IEEE Computer Society, 295–304.

[17] H. Gimbert and Y. Ouahladj. 2010. Probabilistic Automata on Finite Words: Decidable and Undecidable Problems. In *ICALP*. LNCS 6199, 527–538.

[18] Raymond Greenlaw, H. James Hoover, and Walter L. Ruzzo. 1995. *Limits to Parallel Computation: P-completeness Theory.* Oxford University Press, Inc.

[19] D. Grigoriev and N. Vorobjov. 1988. Solving systems of polynomial inequalities in subexponential time. *Journal of symbolic computation* 5, 1-2 (1988), 37–64.

[20] Darald J. Hartfiel. 1998. *Markov Set-Chains.* Springer.

[21] Shizuo Kakutani. 1941. A generalization of Brouwer's fixed point theorem. *Duke Math. J.* 8, 3 (09 1941), 457–459.

[22] O. Madani, S. Hanks, and A. Condon. 2003. On the undecidability of probabilistic planning and related stochastic optimization problems. *Artificial Intelligence* 147, 1-2 (2003), 5–34.

[23] L. Maruthi, I. Tkachev, A. Carta, E. Cinquemani, P. Hersen, G. Batt., and A. Abate. 2014. Towards real-time control of gene expression at the single cell level: a stochastic control approach. In *CMSB*. LNCS/LNBI, 155–172.

[24] Joël Ouaknine, João Sousa Pinto, and James Worrell. 2017. On the Polytope Escape Problem for Continuous Linear Dynamical Systems. In *HSCC'17*. 11–17.

[25] Joël Ouaknine and James Worrell. 2014. Ultimate Positivity is decidable for simple linear recurrence sequences. In *ICALP'14*. Springer, 330–341.

[26] Martin L. Puterman. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming* (1st ed.). John Wiley & Sons, Inc.

[27] Marcus Schaefer. 2009. Complexity of some geometric and topological problems. In *International Symposium on Graph Drawing*. Springer, 334–344.

[28] A. Tiwari. 2004. Termination of linear programs. In *Computer-Aided Verification, CAV (LNCS)*, Vol. 3114. Springer, 70–82.

[29] M. Hirvensalo V. Halava, T. Harju and J. Karhumäki. 2005. Skolem's problem - on the border between decidability and undecidability. In *TUCS Technical Report Number 683*.

[30] Piotr J. Wojciechowski, Pavlos Eirinakis, and K. Subramani. 2014. Variants of Quantified Linear Programming and Quantified Linear Implication. In *ISAIM'14*.

[31] Piotr J. Wojciechowski, Pavlos Eirinakis, and K. Subramani. 2017. Erratum to: Analyzing restricted fragments of the theory of linear arithmetic. *Ann. Math. Artif. Intell.* 79, 4 (2017), 371–392.